Local Search Heuristics for the Teacher/Class Timetabling Problem ⋆

Yuri Kochetov[1], Polina Obuhovskaya[2] and Mikhail Paschenko[1]

[1] Sobolev Institute of Mathematics, Russia
[2] Novosibirsk State University, Russia
obuhovskaya@ccfit.nsu.ru

1 Introduction

We consider the well known NP–hard teacher/class timetabling problem [1]. Variable neighborhood search and tabu search heuristics are developed to find near optimal solutions to this problem. The heuristics are based on two types of solution representation. For each of them we consider two families of neighborhoods. The first family uses swapping of time periods for teacher (class) timetable. The second family bases on the idea of large Lin–Kernighan neighborhoods. Computation results for difficult random test instances show high efficiency of the proposed approach.

2 Problem Formulation

In the teacher/class timetabling problem we are given the following finite sets: J is the set of subjects, K is the set of classes, L is the set of teachers, T is the set of time periods. These periods are distributed in 6 week days. By $T_l \subseteq T$ we denote the set of time periods which are available for teacher l. We suppose that classes are disjoint sets of students, students in a chosen class have the same subjects, and correspondence between subjects and teachers for a chosen class is one–to–one. The number of lessons per week for each class and each teacher is known in advance. We say that a timetable S is feasible if the following requirements are satisfied:
a) a teacher l has at most one lesson at a time period t if $t \in T_l$ and no lessons otherwise;
b) a class k has at most one lesson at a time period t;
c) each teacher must fulfill his (her) weekly number of lessons.

The objective function $F(S)$ is a penalty function for the following soft constraints:
1. each teacher has no time gaps;
2. each teacher has lessons in the most convenient time periods;
3. each class has no double lessons.

⋆ This work was partially supported by Russian Foundation of Basic Research, grant 04–07–900096

E. K. Burke, H. Rudová (Eds.): PATAT 2006, pp. 454–457. ISBN 80-210-3726-1.

More exactly, we wish to minimize the following objective function:

$$F(S) = \sum_{l \in L} \sum_{d=1}^{6} \alpha_{ld} f^1_{ld}(S) + \sum_{l \in L} \sum_{t \in T} \beta_{lt} f^2_{lt}(S) + \sum_{k \in K} \sum_{d=1}^{6} \gamma_{kd} f^3_{kd}(S),$$

where positive α, β, and γ are the penalties and $f^i(S)$ is the number of violations of soft restriction i, $i = 1, 2, 3$. The optimization problem is NP–hard. Moreover, the decision problem on existence of a feasible solution is NP–complete. So, we introduce semifeasible solutions to enlarge the search space and apply metaheuristics for this space to find near optimal feasible solutions.

3 Solution Representations

We introduce two types of semifeasible solutions.

Definition 1. *A timetable S^a is a semifeasible solution of the type a if it satisfies the restrictions b and c.*

Definition 2. *A timetable S^b is a semifeasible solution of the type b if it satisfies the restrictions a and c.*

It is convenient to represent an arbitrary timetable S^a as a $\mathbf{K} \times \mathbf{T}$ matrix (S^a_{kt}), $\mathbf{K} = |K|, \mathbf{T} = |T|$, with values in $\{0, 1, \ldots, \mathbf{J}\}$, $\mathbf{J} = |J|$, where the k-th row is a timetable for the k-th class. Nonzero entries of the row mean subjects for the class k at the time period t; $S^a_{kt} = 0$ means free time. In a similar way we represent S^b as a $\mathbf{L} \times \mathbf{T}$ matrix (S^b_{lt}), $\mathbf{L} = |L|$, with values in $\{-1, 0, 1, \ldots, \mathbf{K}\}$. Entries of the matrix mean classes for the teacher l at the time period t if $S^b_{lt} > 0$, and free time if $S^b_{lt} \leq 0$. The case $S^b_{lt} = -1$ means that the time period t is unavailable for the teacher l. The advantage of this representation is that it eliminates conflicts for teachers. The occurrence of conflicts in column happens when in a given period t more than one teacher is allocated to a class. A solution S^b is feasible if and only if each column has not conflicts. An arbitrary feasible solution S can be easily represented by (S^a_{kt}) and (S^b_{lt}) matrices. In order to evaluate the semifeasible solutions we introduce the following function

$$\overline{F}(S) = F(S) + \sum_{l \in L} \sum_{t \in T} \lambda_{lt} f^4_{lt}(S) + \sum_{k \in K} \sum_{t \in T} \mu_{kt} f^5_{kt}(S),$$

where $\min(\lambda, \mu) > \max(\alpha, \beta, \gamma)$ and $f^4(S)$, $f^5(S)$ are some penalty functions for the restrictions a and b. Obviously, $\overline{F}(S) = F(S)$ if S is a feasible solution. It is easy to realize a transition from a semifeasible solution S^a to S^b and back such that the number of positive items in $\overline{F}(S) - F(S)$ does not increase.

4 Neighborhoods

Now we introduce four families of neighborhoods:

– $N_i(S^a), i \geq 1$, denote swap neighborhoods of a semifeasible solution S^a;
– $N_i(S^b), i \geq 1$, denote swap neighborhoods of a semifeasible solution S^b;
– $LK_i(S^a), i > 1$, denote Lin–Kernighan neighborhoods of S^a;
– $LK_i(S^b), i > 1$, denote Lin–Kernighan neighborhoods of S^b.

The neighborhood $N_1(S^a)$ consists of neighboring solutions which are obtained from S^a by swapping two different values of a given row in the matrix (S^a_{kt}). Each element in this neighborhood is associated with a triplet $\langle k, t', t''\rangle$, where t' and t'' are the time periods, k is the class, and $S^a_{kt'}$ and $S^a_{kt''}$ are the interchanged subjects. For $i > 1$, $N_i(S^a)$ are formed of solutions which are obtained by a sequence of interchanges with triplets $\{\langle k, t'_j, t''_j\rangle\}_{j\leq i}$, $k \in K$ is fixed. Families $N_i(S^b)$ are defined in a similar way. Moreover, only non-negative values of the matrix (S^b_{lt}) can be interchanged. We note that arbitrary feasible solution can be reached with the use of an appropriate sequence of neighboring solutions for the neighborhoods $N_1(S^a)$ or $N_1(S^b)$. A Lin–Kernighan neighborhood $LK_i(S^a)$ consists of i elements and can be described by the following steps [3].

1. Choose a triplet $\langle k, t', t''\rangle$ such that the corresponding neighboring solution $S' \in N_1(S^a)$ is the best even if it is worse than S^a.
2. Put $S^a := S'$.
3. Repeat steps 1, 2 i times; if a triplet was used at steps 1 or 2 of previous iterations, it can not be used any more.

The sequence of triplets $\{\langle k_j, t'_j, t''_j\rangle\}_{j\leq i}$ defines i neighbors S_j of the solution S^a. We say that S^a is a local minimum with respect to the LK_i–neighborhood if $\overline{F}(S^a) \leq \overline{F}(S_j)$ for all $j \leq i$. A local minimum with respect to the LK_i–neighborhood is a local minimum with respect to N_1 and is not necessary a local minimum with respect to N_i, $i > 1$. Family $LK_i(S^b)$ is defined similarly.

5 Variable Neighborhood Search

We adjust the framework of the VNS metaheuristic [2] for our problem as follows.

1. *Initialization.* Find an initial semifeasible solution S; choose a stopping condition and sizes of neighborhood families i_{max}, j_{max}.
2. *Repeat* the following sequence until the stopping condition is met:
   (a) Set $i \leftarrow 1$; if $\overline{F}(S) = 0$ then STOP, return the optimal solution S.
   (b) Repeat the following steps until $i = i_{max}$:
      i. *Shaking*. Generate a solution S' at random from the $N_i(S)$.
      ii. *Local search*. Use a local descent algorithm with respect to N_1 with S' as the initial solution; denote the obtained local minimum as S''.
      iii. *Move or not*. If $\overline{F}(S'') < \overline{F}(S)$ then put $S \leftarrow S''$ and goto 2(a); otherwise, set $i \leftarrow i + 1$.
   (c) i. *Large neighborhood search*. Use a local descent algorithm with respect to neighborhood $LK_{j_{max}}$ with S as the initial solution; denote the obtained local minimum as S''.

ii. Change representation. If $\overline{F}(S'') < \overline{F}(S)$ then put $S \leftarrow S''$, otherwise change the solution representation; goto 2(a).

At the initial step 1 we generate S by a polynomial time heuristic. It has $\mathbf{T}$ stages. At each stage we solve an assignment problem.

6 Computational Results

We test the VNS algorithm on random instances with $\mathbf{T} = 6 \times 5$ and $\alpha = 1 + \alpha',\ \beta = 3 + \beta',\ \gamma = 5 + \gamma',\ \lambda = 10 + \lambda',\ \mu = 10 + \mu'$, where $\alpha', \beta', \gamma', \lambda', \mu'$ are random noise, $0 < \alpha', \beta', \gamma', \lambda', \mu' \ll 1$. This rule removes plateaus and improves the landscape for local search methods. Each class has $\mathbf{T}$ lessons. Each teacher $l \in L$ has $\mathbf{T}_l/5$ inconvenient time periods. The VNS algorithm produces 50 **KT** moves from a solution to a neighboring one. Table 1 presents average values of the objective function for the best found solutions in 50 trials. Each row of the table corresponds to one instance. For all instances VNS finds feasible solutions in all trials. For comparison, we present the results for a tabu search method with and without changing the solution representation (columns TSR and TS). Table 1 shows that change of the solution representation is a useful idea for both methods. We hope it may be successfully applied for other approaches as well.

n	$\mathbf{L}$	$\mathbf{K}$	$\mathbf{J}$	$\sum_l \mathbf{T}_l$	TS	TSR	VNS	VNSR
1	14	6	83	210	32.9	27.7	30.3	27.0
2	16	8	120	336	21.6	15.7	17.9	15.1
3	23	12	195	552	82.3	67.5	65.5	64.2
4	31	13	207	558	70.1	66.9	67.4	66.3

Table 1. Average values of $\overline{F}(S)$

References

1. Garey, M.R., Johnson, D.S.: Computers and intractability: a guide to the theory of NP–completeness. Freeman, San Francisco, CA, (1979)
2. Hansen, P., Mladenović, N.: Developments of Variable Neighborhood Search. In Ribeiro, C.C., Hansen, P.(Eds): Essays and surveys in metaheuristics. Kluwer Academic Publishers, (2002)
3. Kochetov, Y., Alekseeva, E., Levanova, T., Loresh, M.: Large neighborhood local search for the p–median problem. Yugoslav Journal of Operations Research **15** (2005), Number 1, 53-63